

## Data format sd\_liste\_rela

---

**Summarized:**

## Contents

---

<a href="#">1 General information.....</a>	<a href="#">3</a>
<a href="#">2 Tree structure.....</a>	<a href="#">3</a>
<a href="#">3 Contents of objects.....</a>	<a href="#">3.3.1</a>
<a href="#">Object “.RLNR”.....</a>	<a href="#">3.3.2</a>
<a href="#">Object “.RLTC”.....</a>	<a href="#">4.3.3</a>
<a href="#">Object “.RLTV”.....</a>	<a href="#">4.3.4</a>
<a href="#">Object “.RLNT”.....</a>	<a href="#">4.3.5</a>
<a href="#">Object “.RLPO”.....</a>	<a href="#">4.3.6</a>
<a href="#">Object “.RLCO”.....</a>	<a href="#">4.3.7</a>
<a href="#">Object “.RLNO”.....</a>	<a href="#">4.3.8</a>
<a href="#">Object “.RLDD”.....</a>	<a href="#">4.3.9</a>
<a href="#">Object “.RLBE”.....</a>	<a href="#">5</a>
<a href="#">3.10 Object “.RLSU”.....</a>	<a href="#">5</a>

## 1 General information

---

the data structure `sd_liste_rela` are a “volatile” data structure which is useful in operators of structure intermediate between the data of the user (linear relations with dualiser) and their translation in finite elements of Lagrange (in the `LIGREL` of `CHARGE`) and in `CARDS` containing the coefficients of relations.

Approximately, the idea is the following one: one progressively stores the linear relations (one by one) in a `sd_liste_rela` (routine `AFRELA`) then one “pours” the `sd_liste_rela` in the `sd_char_xxx` at the end of the command (routine `AFLRCH`).

**Note:**

one can create several `sd_liste_rela` within the same command `AFFE_CHAR_XXX`. It is then necessary “to pour” each one of these SD in the `sd_char_xxx`. That makes it possible to make independent the various routines which manage linear relations: `caddli`, `cafaci`,...

## 2 Tree structure

---

```
sd_liste_rela (K19)  :: =record
(O) ".RLCO": OJB   S V R/C
(O) ".RLBE": OJB   S V R/C/K24
(O) ".RLDD": OJB   S V K8
(O) ".RLNO": OJB   S V K8
(O) ".RLNT": OJB   S V I
(O) ".RLPO": OJB   S V I
(O) ".RLSU": OJB   S V I
(O) ".RLTC": OJB   S.E. K8
(O) ".RLTV": OJB   S.E. K8
(O) ".RLNR": OJB   S.E. I
```

## 3 Contained Notations

---

**objects:**

- `nb_rela`: many relations stored in the `sd_liste_rela`.
- `nb_coef_lrel`: many coefficients of a linear relation (“left” part of the equation). The coefficients can be: realities or complexes.
- `nb_coef_tot`: nombre total of coefficients of all the linear relations stored in the `sd_liste_rela`.  
$$\text{nb\_coef\_tot} = \text{nb\_coef\_lrel} (1) + \text{nb\_coef\_lrel} (2) + \dots$$
- `coef_impo`: specified value for the linear relation (“right” part of the equation or “second member”). The specified value can be: real, complex or function.

### 3.1 Object “.RLNR”

This object is length 1.

`RLNR (1)`: many linear relations `nb_rela`.

## 3.2 Object ".RLTC"

This object is length 1.

RLTC (1) : type of the coefficients of relations "REEL" or "COMP".

## 3.3 Object ".RLTV"

This object is length 1.

RLTV (1) : type of the second members of relations "REEL", "COMP" or "FONC".

## 3.4 Object ".RLNT"

This object is length nb\_rela.

It contains the numbers of coefficients for each relation.

RLNT (irela) : many coefficients of the relation irela = nb\_coef\_1rel (irela).

## 3.5 Object ".RLPO"

This object is length nb\_rela.

It makes it possible "to point" (for a given relation) in objects .RLCO, .RLNO and .RLDD.

These 3 objects have same dimension and one points there in the same way.

RLPO (irela) : addresses in .RLCO (for example) last coefficient of the relation irela.

RLCO (RLPO (irela) - RLNT (irela) + 1) is the 1st term of the relation irela.

## 3.6 Object ".RLCO"

This object is length nb\_coef\_tot.

It R contains the coefficients ( or C) of the equations.

## 3.7 Object ".RLNO"

This object is length nb\_coef\_tot.

It contains the names of the nodes implied in the relations.

## 3.8 Object ".RLDD"

This object is length nb\_coef\_tot.

It contains the names of the CMPS implied in the relations.

## 3.9 Object ".RLBE"

This object is length `nb_rela`.

It contains the second members of the relations.

```
RLBE (irela) : second member of the relation irela = coef_impo (irela).
```

## 3.10 Object ".RLSU"

This object is length `nb_rela`. It is an indicator to say if the linear relations must be taken into account (or not).

Indeed, before "pouring" the linear relations in the `sd_charge`, one examines whether certain relations are not given in several specimens (duplicates) or if those should not be eliminated (`METHODE='ELIMINATION'`).

```
RLSU (irela) :  
/0 - > the relation irela is to be taken into account.  
/1 - > the relation irela is the duplicate of a preceding relation.  
       It should not be taken into account  
/2 - > the relation irela must be eliminated (METHODE=' ELIMINATION').  
       It should not be taken into account
```